Lighthouse Fails Your Llms.txt Without Markdown Links via @sejournal, @slobodanmanic

Chrome’s new Lighthouse Agentic Browsing audit treats your .txt file as a markdown document. If your llms.txt does not use markdown link syntax, you fail the audit, even when every link in the file is accurate and works. I ran the audit on nohacks.co. Two of six audits passed. Three came back not applicable. One failed: the llms.txt audit, with the verbatim error *“File does not appear to contain any links.”* The fix was five characters per link. The file is still served as plain text. Only the audit result changed.

Lighthouse 13.3.0 shipped the Agentic Browsing category alongside Performance, Accessibility, SEO, and Best Practices. Six audits in the default set: accessibility tree well-formedness (agent-accessibility-tree), cumulative layout shift (cumulative-layout-shift), llms.txt discoverability (llms-txt), and three WebMCP checks (webmcp-registered-tools, webmcp-form-coverage, webmcp-schema-validity). The category returns a fractional pass ratio instead of a 0-to-100 score, because the standards for the agentic web are still in motion.

1 Of 6 Audits Failed On Nohacks.co

I ran the audit via the Lighthouse CLI: npx lighthouse@latest https://nohacks.co --only-categories=agentic-browsing. Six audits returned. Three came back not-applicable, all WebMCP: webmcp-registered-tools, webmcp-form-coverage, and webmcp-

schema-validity. Lighthouse gives no reason for a not-applicable result, it just marks the audit and moves on. nohacks.co does expose WebMCP, but only through the experimental imperative navigator.modelContext API (two glossary tools, two for an agentic-browser directory), with no declarative form annotations. The scan ran in a default headless Chrome 150 with no WebMCP flag, so the not-applicable verdict could mean the website exposes nothing these audits recognize, or that the scan environment had no WebMCP API active at the time. Lighthouse does not say which. Two audits passed cleanly: agent-accessibility-tree reported *“All audits passed,”* confirming the semantic HTML and ARIA structure is well-formed enough for agents to navigate, and cumulative-layout-shift came back at zero.

One audit failed: llms-txt. The verbatim error message from Lighthouse was:

> *File does not appear to contain any links.*

The category score was 0.67. That was the first surprise. The file at nohacks.co/llms.txt has many links. Navigation paths to articles, episodes, guests, the glossary. RSS feed URLs. Audio file URL patterns. The file is over five kilobytes of structured content. So why was Lighthouse reporting zero links?

Lighthouse Parses .txt As Markdown And Rejects Plain-Text Links

The file extension is .txt, but Lighthouse parses the contents as markdown, and demands markdown link syntax for any text to count as a link. The file is named llms.txt. The HTTP server returns it with a text/plain MIME type. Open it in a browser, and you see plain text. But the llms.txt specification at llmstxt.org defines the format as a markdown document. The

spec is explicit: *“Each section contains a markdown bullet list of links. Each list item has a link followed by optional notes about the link, separated from the link by a colon.”* Lighthouse’s parser enforces that strictly. Every link must be encoded as markdown link syntax, [text](url), with square brackets around the link text and parentheses around the URL.

My file had been using a more natural plain-text format:

```
- Homepage: / - Publication masthead, cornerstone series, latest articles and episodes
- Articles: /blog - All articles on AXO, the agentic web, and AI agents
- Episode: /episode/[slug] - Full show notes, transcript, audio player
```

Same destinations. Same descriptions. Same information. Lighthouse’s parser does not register those lines as links. Across the entire file, it registered exactly zero. Audit fails.

A file with a .txt extension, served with a text/plain MIME type, that fails an audit unless it is formatted as markdown. That is a mismatch the audit layer is going to have to be more honest about. The extension says one thing. The MIME type says one thing. The parser is the source of truth, and the parser demands markdown.

The Fix Is 5 Characters Per Link

Wrap each link target in markdown bracket-paren syntax, [text](url), and replace the - separator before each description with : . Five characters per link. Mechanical conversion, repeated across the file.

```
- [Homepage](/): Publication masthead, cornerstone series, latest articles and episodes
- [Articles](/blog): All articles on AXO, the agentic web, and AI agents
- [Episode](/episode/[slug]): Full show notes, transcript,
```

audio player

I made the edit. Re-ran the audit. Score went from 0.67 to 1.0. The audit title flipped from *“llms.txt does not follow recommendations”* to *“llms.txt follows recommendations.”* No detail items in the after-report. Clean pass.

The file is still served as text/plain. The file extension is still .txt. The file content is still the same content. Only the link encoding changed.

Lighthouse Measures Parseable Link Syntax, Not File Quality

The audit checks whether your file is mechanically parseable. It does not check whether the file describes your website usefully. Both reads are true at the same time.

The first read: The audit is measuring something real. Markdown link syntax is mechanically parseable. Plain-text descriptive lines are not. If an AI agent (or the Lighthouse parser standing in for an agent) needs to extract links from the file programmatically, the markdown format is required. The audit is correct that the file before my fix could not be parsed for links by the standard tooling. The conversion to markdown link syntax fixes a real interoperability gap.

The second read: format compliance is not the same as file quality. A thoughtfully-written, accurate, comprehensive llms.txt that uses plain-text descriptions fails this audit. A thin, auto-generated llms.txt with markdown link syntax passes. The audit cannot tell the difference between the two. The WordPress plugin AIOSEO, used by over 3 million websites per its WordPress.org listing, generates llms.txt files with markdown link syntax by default, a default-on behavior Glenn Gabe surfaced, and the plugin’s own documentation confirms. Those auto-generated files use markdown link syntax because that is what the generator emits. Most of them probably pass

this audit. Most hand-curated, owner-aware llms.txt files probably fail it.

That gap is worth thinking about before treating the audit's pass/fail as a measurement of how agent-ready your website really is. The audit is checking whether your file is parseable. It is not checking whether your file is useful.

Should You Care About Lighthouse Agentic Browsing's Llms.txt Check?

Yes, but narrowly. Lighthouse can tell you whether your llms.txt is parseable as markdown. It cannot tell you whether the file describes your website honestly. That check is yours. Open Chrome DevTools, click the Lighthouse tab, verify the Agentic Browsing category is checked, and run Analyze on your URL. The audit takes under a minute. If it fails on the no-links error, the fix is five characters per link and five minutes of editing. If it passes, the harder question is the one Lighthouse cannot ask. Was the file auto-generated by a plugin you did not configure, or did you write it yourself, and either way, does it describe what your website actually is?

The Machine-First Architecture **Structure pillar** sits beneath all of this: data models before page layouts, rendering independence, content that does not depend on client-side JavaScript or human-display defaults to be machine-readable. The llms.txt audit is a narrow check at that layer. The bigger structural question, whether your machine-readable surface describes your website accurately, is yours to run.

More Resources:

This post was originally published on No Hacks.

Featured Image: Darko 1981/Shutterstock

https://www.searchenginejournal.com/lighthouse-fails-your-llms-txt-without-markdown-links/577590/